

5. Übung: Nichtlineare Nachbarschaftsfilter

Michael Grunwald

Aufgabe 1

- Aufgabe 1.1 Implementieren Sie den Medianfilter (MEX Funktion (C/C++)).
 - Prototyp: `[out_image] = mxMedianFilter(in_image, filter, off);`

`out_image` Ergebnisbild nach Operation von `in_image` mit `filter`
`in_image` Eingangsbild (uint8)
`filter` Filtermatrix (double)
`off` Offset (uint8)
 - 8-Bit Graustufenbilder als Eingangs- und Ausgangsdatum.
 - Filtermatrix der Größe (N×N) mit $N = (2K + 1)$, $K = 1, 2, \dots$
 - Ankerpunkt (Hot Spot) ist die Mitte der Filtermatrix.
 - Benutzung Sie für die Sortierung – wenn möglich – Heap Sort.
 - Beispielaufwurf: `im = mxFilter(imread('bild.jpg'), [1 1 1; 1 3 1; 1 1 1]./11, uint8(0));`
- Aufgabe 1.2 Laden Sie das Buch Digitale Bildverarbeitung herunter und lesen Sie Kapitel 6.4 (*Nichtlineare Filter*) und Kapitel 7 (*Kanten und Konturen*) . Bitte beachten Sie, dass der Link nur intern im HTWG Netz funktioniert.

Aufgabe 2

- Aufgabe 2.1 Warum ist es sinnvoll für die Sortierung Heap Sort zu verwenden?
- Aufgabe 2.2 Untersuchen Sie, welche für Effekte bei mehrmaligem Anwenden eines Filters auf das jeweilige Ergebnisbild auftreten. Welche Effekte treten bei großen und bei kleinen Filtermasken auf?

Bemerkung

- Bitte erstellen Sie für die verwendeten cpp-Dateien die zugehörigen kommentierten Matlab m-Dateien.
- Die Aufgaben werden elektronisch als **eine** komprimierte Datei (m.grunwald@htwg-konstanz.de) und per Ausdruck abgegeben. (Ausdruck: mxMedianFilter.cpp und Beantwortung der Aufgabe 2, Elektronisch: Alle cpp-Dateien und Aufgabe 3 als pdf-Datei)